



ваются в формате XML, логико-временная схема так же интерпретируется в XML-структуру. Это позволяет использовать технологию генерации с помощью XSLT (языка преобразования XML). XSLT имеет массу преимуществ: выполнение повторной генерации кода без перекомпиляции приложения (необходимо только изменить шаблон); возможность генерации программы на любом языке программирования; использование наглядного и широко распространенного XML формата [3].

Разработанный модуль генерации является частью программного комплекса ГРАФКОНТ, разрабатываемого по заказу ГНПРКЦ "ЦСКБ-ПРОГРЕСС" на кафедре программных систем СГАУ. Задача системы ГРАФКОНТ состоит в том, чтобы автоматизировать процессы проектирования, создания, документирования и тестирования бортовых управляющих алгоритмов и программ реального времени для космических аппаратов, обеспечить повышение качества и надежности программ, а так снизить трудоемкость и стоимость разработки.

Литература

1. Калентьев, А.А., Тюгашев, А.А. ИПИ/CALS технологии в жизненном цикле комплексных программ управления [Текст] / А.А. Калентьев, А.А. Тюгашев – Самара: Изд-во Самарского научного центра РАН, 2006. – 285 с.
2. Трусов, В.С. Система визуального конструирования временных диаграмм управляющих алгоритмов беспилотных ЛФ [Текст]: дисс. к. т. н./ Трусов Виталий Сергеевич // Самарский государственный аэрокосмический университет им. акад. С.П. Королева – Самара, 2005. – 116 с.
3. Канжелев, С.Ю. Автоматическая генерация автоматного кода [Текст] / С.Ю. Канжелев, А.А. Шалыто // Информационно-управляющие системы / Санкт-Петербургский государственный университет аэрокосмического приборостроения. – 2006. – С.35-42.

С.П. Орлов, Е.Ю. Биктимиркин, А.А. Тютнев

ИМИТАЦИОННАЯ МОДЕЛЬ МНОГОПОРТОВОЙ ПАМЯТИ МИКРОПРОЦЕССОРОВ

(Самарский государственный технический университет)

Современные микропроцессорные системы используют многопортовую (многовходовую) память для исключения конфликтов при параллельном или конвейерном выполнении потока команд. При этом возникает задача предотвращения конфликтов внутри самой многопортовой памяти, так как обращения от нескольких операционных устройств производятся к одной и той же ячейке памяти. На рис. 1 приведена схема 6-ти транзисторной ячейки SRAM памяти [1]. Здесь имеется N адресных шин B_i и N шин данных W_j , что соответствует N -входовой памяти.

Для обеспечения многопортового доступа в такую ячейку применяются следующие механизмы разрешения конфликтных ситуаций:

- арбитражная логика;
- семафоры;
- запросы на прерывания.

В докладе описывается имитационная модель, которая позволяет изучать процессы обращения к многопортовой памяти, способы разрешения конфликтов, исследовать характеристики производительности, длины очередей запросов и другие параметры.

Имитационная модель построена на базе сети Петри [2]. Такой подход целесообразен, так как сеть Петри моделирует события в объекте.

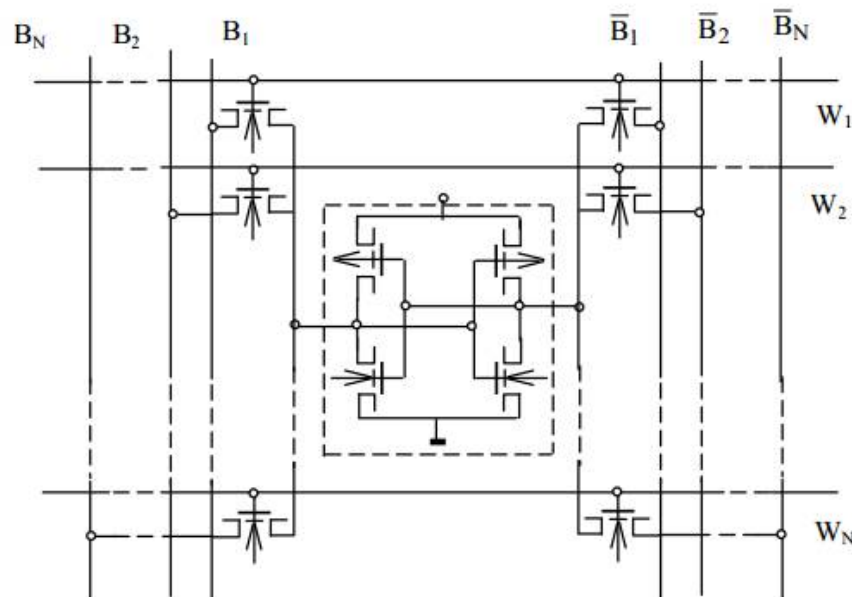


Рис. 1. Многопортовая шеститранзисторная ячейка памяти

Для учета временных соотношений в моделируемой системе памяти будем использовать подкласс сетей Петри - временные сети C^t [3]:

$$C^t = (P, T, U, \mu_0, \beta),$$

где P – конечное множество позиций, T – конечное множество переходов, $U \subseteq P \times T \cup T \times P$ – отношение инцидентности, μ_0 – начальная разметка в начальный момент времени, $\beta: P \rightarrow R$ – отображение, связывающее позиции с временами их готовности, т.е. маркер в позиции p_i становится готовым только через время τ_i , после того, как он появился в позиции.

Основным компонентом имитационной модели служит фрагмент временной сети Петри с двумя источниками запросов [4]. Это позволяет исследовать двухпортовую память и достаточно легко распространить методику на 2-х, 4-х и 8-ми портовые ячейки памяти. На рис. 2 приведен такой фрагмент сетевой модели.

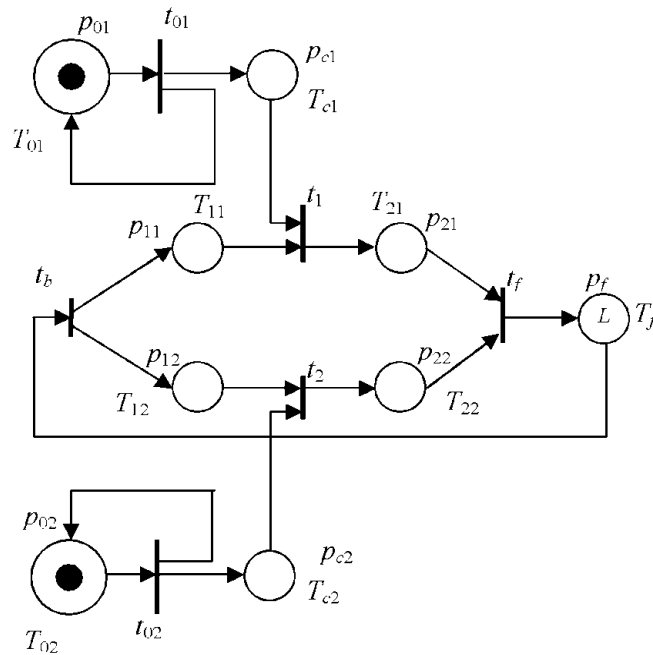


Рис. 2

Позиции и переходы интерпретируются следующим образом.

- позиции p_{01} и p_{02} , переходы t_{01} и t_{02} моделируют поток запросов на доступ к ячейке памяти;
- позиции p_{c1} и p_{c2} синхронизируют обращение к памяти в соответствии с выбранной логикой разрешения конфликтов;
- позиции p_{11} , p_{12} , p_{21} , p_{22} и переходы t_1 , t_2 образуют внутреннюю структуру ячейки памяти;
- позиция p_f моделирует информационный ресурс ячейки с L ресурсными единицами (маркерами), причем на эту позицию не распространяется требование безопасности;
- маркеры (темные точки) в позициях моделируют поступления сообщений.

В зависимости от частоты поступления маркера от источников запросов и соотношения временных параметров параллельных ветвей в ячейке памяти может произойти накопление маркеров в p_c и образование очереди сообщений. Следовательно, для моделируемой сети надо определить: условие L – ограниченности для ресурсной позиции, условия безопасности сети для остальных позиций, допустимую длину очереди.

Для модели на рис. 2 найдено рекуррентное выражение для времени срабатывания переходов:

$$\begin{aligned}\tau_{n+1}(t_b) &= T_f + \max\{T_{21} + \max\{(n-1)T_{01} + T_{c1}\}, \tau_n(t_b) + T_{11}\}, \\ T_{22} + \max\{(n-1)T_{02} + T_{c2}, \tau_n(t_b) + T_{12}\}\}, \\ \tau_1(t_b) &= 0.\end{aligned}\quad (1)$$

где $T_f, T_{c1}, T_{c2}, T_{01}, T_{02}, T_{11}, T_{21}, \dots$ - времена, приписанные соответствующим позициям сети.



Отсюда получаем условия безопасности для синхронизирующей позиции рс:

$$\begin{aligned}\tau_n(t_b) + T_{11} &\leq nT_{01}, \\ \tau_n(t_b) + T_{12} &\leq nT_{02}.\end{aligned}\tag{2}$$

Для различных временных соотношений в памяти с помощью формул (1) и (2) можно найти длину очереди сообщений и число сообщений в потоке, не нарушающих безопасность сети. В модели также реализована возможность задавать случайные последовательности запросов с выбранным законом распределения. Очевидно, что описанная методика может быть распространена на произвольное число параллельно выполняющихся процессов запросов к многопортовой памяти.

Предлагаемая модель позволяет исследовать различные структуры многопортовой памяти, определить эффективность механизмов разрешения конфликтов.

Литература

1. Лементуев, В.А. Многопортовая память микропроцессорных систем/В.А.Лементуев//Информационные технологии и вычислительные системы, № 2, 2009. – С. 96 – 101.
2. Питерсон, Дж. Теория сетей Петри и моделирование систем/ Дж. Питерсон. – М.: Мир, 1984. – 264 с.
3. Orlov S.P. Application of Petri net model for computational process synchronization/ S.P. Orlov// Advances in Modeling & Analysis. Vol.14. №3.-AMSE PRESS,1993. –P.1-6.
4. Орлов, С.П. Сеть Петри для моделирования межпроцессорного обмена в вычислительном кластере/С.П.Орлов// Материалы Всеросс. научн.-технич. конф. «Информационные технологии в науке и производстве (ИТНП-2013)» - Самара: СамГТУ, 2013. – С.207-210.

А.И. Пугачев

МОДЕЛЬ ПРОИЗВОДСТВА И РАСПРЕДЕЛЕНИЯ РЕСУРСОВ

(Самарский государственный технический университет)

Функционирование предприятий, имеющих подразделения вспомогательных производств, основано не только на расходовании различных видов внешних ресурсов, но также и на выработке и внутреннем потреблении собственных ресурсов [1]. В итоге сложные внутрипроизводственные маршруты движения собственных ресурсов создают серьезную проблему как в расчете их себестоимости, так и, в конечном счете, в расчете себестоимости готовой продукции.

Рассмотрим производственную систему, в составе которой имеется m